Department of Computer Science and Engineering
The University of Texas at Arlington

Team: Door Keepers

Project: Smart Garage

Team Members:
Anup Patel
Santosh Shrestha
Wasyhun Tesfaye
Adrian Echavarria

Last Updated: Wednesday, August 6, 2014, 8:53 PM

Table of Contents

Table of Contents....2
Document Revision History....5
List of Figures....6
List of Tables....6
1 Product Concept....7
1.1 Purpose and Use....7
1.2 Product Layout....7
1.3 Intended Audience....8
2 Product Description and Functional Overview....9
2.1 Features and Functions....9
2.2 External Inputs and Outputs....9
2.3 Product Interfaces....10
3 Customer Requirements....12
3.1 Door Status....12
3.2 Open Garage Door....12
3.3 Close Garage Door....13
3.4 Turn Lights On....13
3.5 Turn Lights Off....13
3.6 Security Notification....14
3.7 System Log Record....14
3.8 Unit Operating Alert System....14
3.9 Multiple Garage Doors....15
3.10 Intercom Communication....15
3.11 Storing Video Surveillance....15
3.12 Live Video Feed....16
3.13 Two-Step Verification....16
3.14 Notification Center....17
3.15 Time Period Security System....17
3.16 Mobile Application Control....17
3.17 Turn On/Off Notifications....18

3.18 Turn On/Off Buzzer 18
3.19 Clear Log Records 18
4 Packaging Requirements 20
4.1 The Recovery Disc 20
4.2 Parts Listing 20
4.3 Full User Manual 21
4.4 Quick User Guide 21
4.5 Smartphone Application 22
5 Performance Requirements 23
5.1 Garage Response Time 23
5.2 Status Update Time 23
5.3 Camera 23
6 Safety Requirements 25
6.1 Unit Installation 25
6.2 Electrical Safety 25
7 Maintenance and Support Requirements 26
7.1 User Manual 26
7.2 Installation/System Setup 26
7.3 Hardware Support/Upgrades 27
7.4 Software Support/Upgrades 27
7.5 Source Code 27
8 Other Requirements 29
8.1 Customer Facing Interface 29
8.2 Temperature Tolerance 29
8.3 Customer Device Account 30
8.4 Hardware System Setup 30
8.5 Software System Setup 30
8.6 Garage Door Open and Close Time 31
8.7 Browser Support 31
9 Acceptance Criteria 32
9.1 The system must be able to update and display the status of the garage door. 32
9.2 The system must have control operations for the garage door over the Internet. 32

9.3 The system must have control operations for the Smart Garage's light over the Internet. 32
9.4 The team must deliver a mobile application to control the garage door. 33
9.5 The system must produce a notification when the garage door stays open for more than a fixed period of time. 33
9.6 The system must produce a notification when an attempt is made to open the garage door without the signal sent from the unit. 33
9.7 The system must provide live video of the garage door. 34
9.8 The system must store the events occurring in the Smart Garage system and display them through the website or mobile app. 34
9.9 The system must ask for security code for performing certain activities. 34
9.10 The system must sound a buzzer when the door is triggered from the unit. 35
9.11 The system must provide the option to turn on or off notifications. 35
9.12 The system must provide the option to turn on or off the buzzer. 35
9.13 The system must provide an option to clear the log record. 35
10 Use Cases 36
10.1 Smart Garage App Use Cases 36
11 Feasibility Assessment 39
11.1 Scope Analysis 39
11.2 Research 39
11.3 Technical Analysis 40
11.4 Cost Analysis 41
11.5 Resource Analysis 41
11.6 Schedule Analysis 42
11.7 Conclusion 46
12 Future Ideas 47
12.1 Customer Requirements 3.10: Intercom Communication 47
12.2 Other Requirements 8.2: Temperature Tolerance 47
12.3 Customer Requirements 3.11: Storing Video Surveillance 47
12.4 Customer Requirements 3.9: Multiple Garage Doors 47

Document Revision History

Revision Number	Revision Date	Description	Rationale
0.1	2/23/2014	First Draft	Checking Progress of SRS
0.2	2/25/2014	Completed Draft	Finalized Adding Sections
0.3	2/25/2014	Fixing Sections/Missing Info	Easier to see whole picture now.
~~**1.0**~~	2/27/2014	Draft Turn In	Due for submission.
0.4	3/5/2014	Draft Fixes	As suggested from Mr. O'Dell
0.5	3/18/2014	Draft Restructuring	Refresh the document and retouch all sections.
0.6	3/25/2014	Incorporating Fixes	Peer Review suggestions to fix issues.
0.6.1	3/26/2014	Minor Fixes	Last minute fixes and rewordings.
1.0	3/27/2014	Final Review	Final review before submission
2.0	4/7/2014	Baseline	Final revisions before baseline submission.
2.1	8/6/2014	Baseline Update	Revisions and updates

List of Figures

Figure 1: Product Concept 8
Figure 2: App Screen Mockups 11
Figure 3: Smart Garage Use Case Diagram 36

List of Tables

Table 1: External Inputs and Outputs 10
Table 2: Cost Estimation 41
Table 3: Function Points 43
Table 4: Influence Multipliers 44

1 Product Concept

This section describes the idea of our product and its general use, features, and audience.

1.1 Purpose and Use

The general concept of this product is to give a homeowner the ability to monitor and control their home's garage door via a smartphone application and an online interface. Homeowners will be able to remotely open, close, and visually see their garage door. This also gives peace of mind to the homeowner that they shut the door after leaving their house. Future expansion includes controlling more than one garage door, intercom and video surveillance of the garage, to help give better security and ease of use.

1.2 Product Layout

Our product in general is focused primarily around the microcontroller, which serves as the brains to our Smart Garage Unit. The microcontroller will need to be connected to the home wireless system which allows the homeowner to access the device remotely. The microcontroller will output commands from the homeowner directly to the garage door opener. The microcontroller also has its own external components such as a camera, door sensors, and the power source. Internally, the microcontroller has components such as a database and file storage. Lastly, the microcontroller is capable of future expansion to a home security system or even an intercom that will allow someone outside the home to communicate with the homeowner.

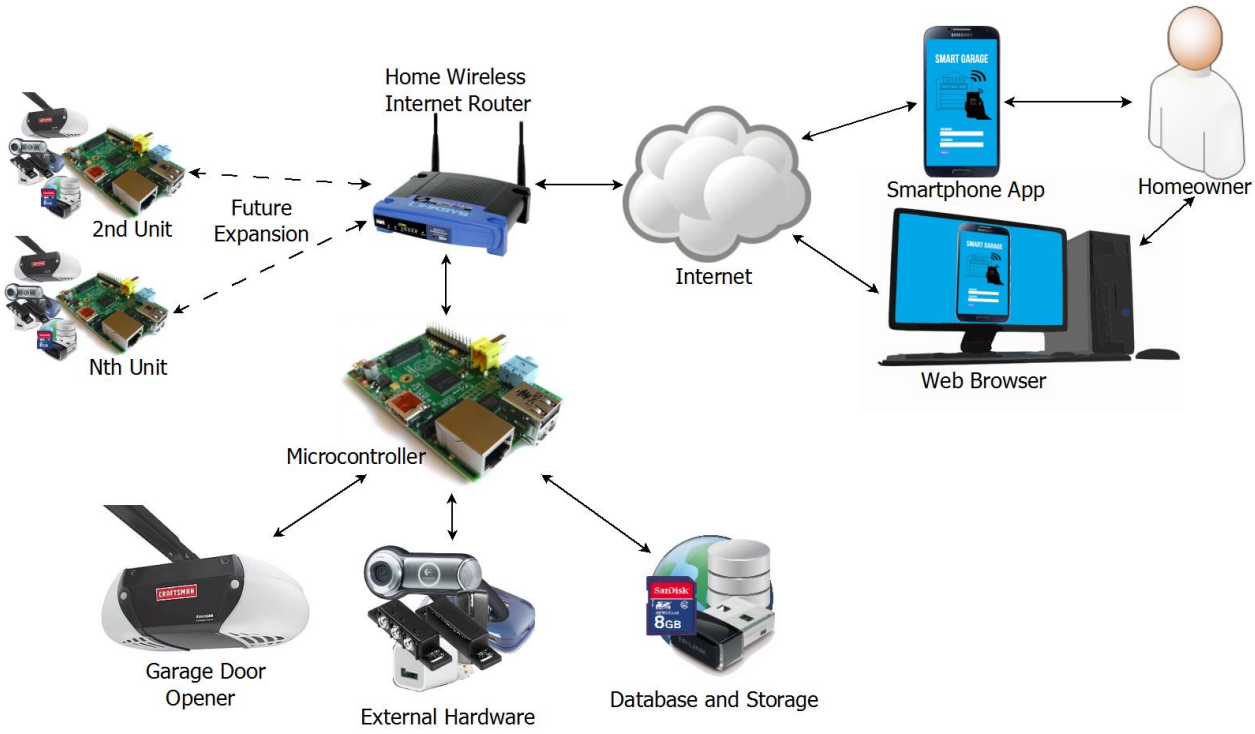

Figure 1: Product Concept

1.3 Intended Audience

Any homeowner with a smartphone, a home computer, an existing electronic garage door opener, some tech know-how, and a home internet connection would be our ideal audience. The homeowner would need at least some tech-capability as the microcontroller would need to be hooked into their existing electronic garage door opener and certain settings would require setup.

2 Product Description and Functional Overview

The primary function of our product is to provide control to a homeowner to open or close their garage door remotely. The product will provide multiple forms of interaction as defined in the remainder of this section.

2.1 Features and Functions

Our product will allow a user to extend their existing garage door opener unit with a Wi-Fi based Internet accessory. The unit will have a few major components: a microcontroller, web camera, door sensors, a web page for setup, and a smartphone app for control. The microcontroller will function as the brains of the Smart Garage product. It will accept user input from the web page, smartphone application and data from the door sensors. After a command is sent, the microcontroller will spur the garage door opener into action, opening or closing the door. The microcontroller will also be able to check the status of the garage door, whether it is open or closed, and send pictures from the included camera. The microcontroller will contain an embedded web server for basic login, command issuances, settings changes, and live video feed.

2.2 External Inputs and Outputs

Name	Type	Description	Use
Garage Door Sensor	Input	Tells the micro-controller whether the garage is open or closed.	If the door is open, the sensor link will be broken and the microcontroller will receive a status that says the door is open.
Garage Door Opener	Output	Opens and closes the garage door	Receives commands from our microcontroller to open or close the garage door.
Web and Mobile Application Login	Input	A webpage with a simple login screen asking for username and password.	Allows the homeowner to gain access to the web-server on the microcontroller and perform operations.
Web and Mobile Application Home Screen	Output	A webpage with sections including status of garage door and commands to perform.	Allows the homeowner to operate their garage door, such as opening the door or turning on light, as well as see the current status of the door.
Web and Mobile Application Open Door	Input	A webpage with a single user entry field that asks the user for a 4 digit code specially marked on the garage door unit for safety.	If the user wishes to open the door remotely, they must input the special 4 digit code that is hardcoded into the software in the microcontroller.

Web and Mobile Application Status	Output	A webpage that will list the status of the garage.	Gives the homeowner the current status of their garage door.
Web and Mobile Application Close Door	Input	A webpage with a single user entry field that asks the user for a 4 digit code specially marked on the garage door unit for safety.	If the user wishes to close the door remotely, they must input the special 4 digit code that is hardcoded into the software in the microcontroller.
Web and Mobile Application Light Control	Input	A webpage that will turn on or off the light in the garage	Allows the homeowner to turn on or off the light in the garage.
Web and Mobile Application Logout	Output	A webpage that will log the user out.	Allows the homeowner to log out of their system so no one can get into it.
Web and Mobile Application Update Status	Output	A webpage that will refresh the status of the garage.	Keeps the homeowner updated on their garage door after sending a command.
Alarm Buzzer	Output	A Buzzer in the unit will go off when triggered.	Buzzes loudly to let anyone inside or near the garage know about garage door activity.
Camera	Input	A camera will be attached to the Smart Garage for a homeowner to remotely check the stats of their door.	Serves as a secondary verification of the system to allow the homeowner to visually see their garage door.

Table 1: External Inputs and Outputs

2.3 Product Interfaces

We will design our mobile application to mimic the functionalities of the web-page. First, we will design our webpage to deliver functionality to the Smart Garage. Then we will develop the android application to match the webpage interaction exactly. The following mockups will be for an android application, but the same will be formatted to be responsive for laptop or desktop pc.

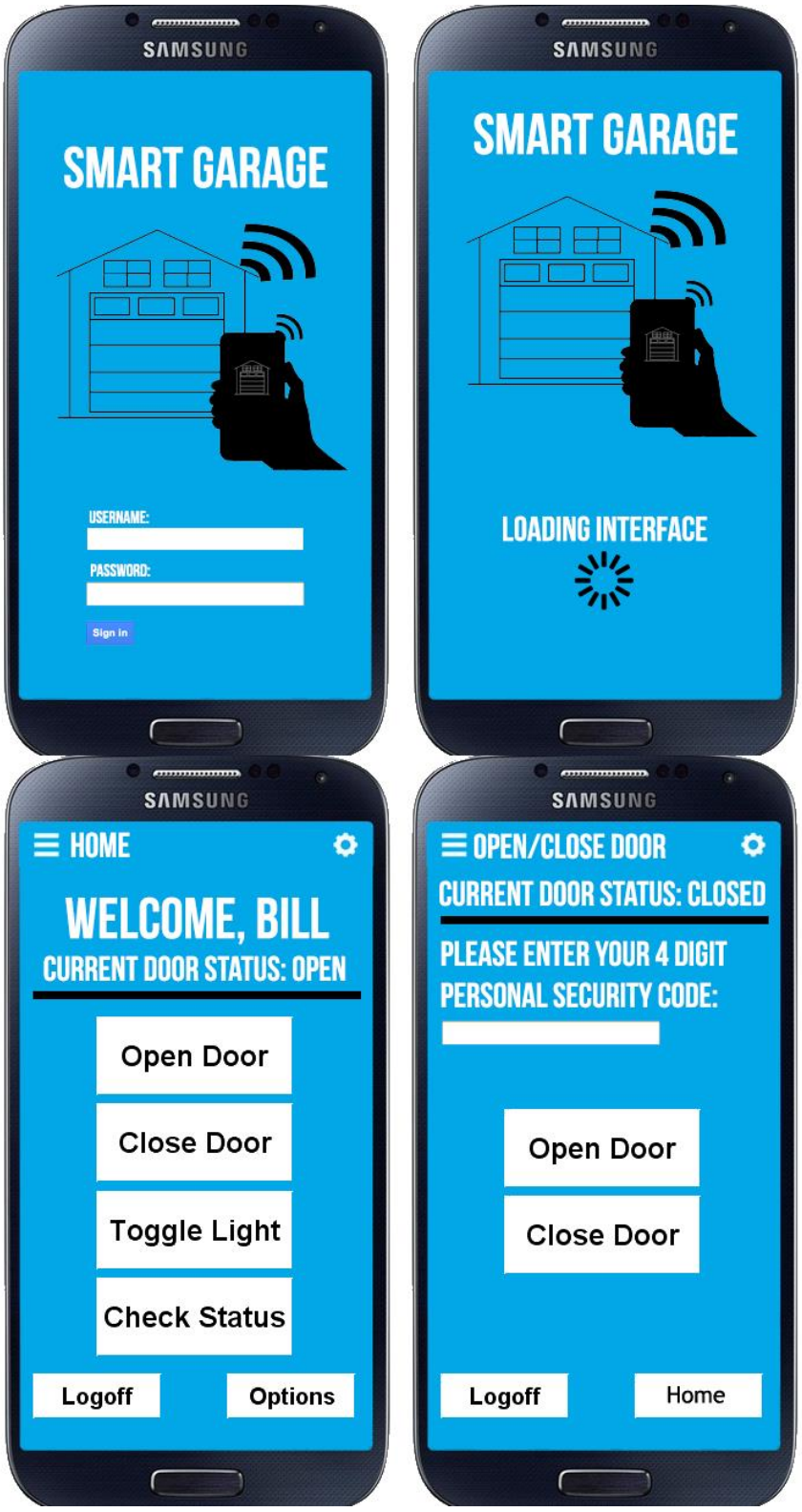

Figure 2: App Screen Mockups

3 Customer Requirements

This section covers the requirements that are required features and functions for our project which were specified by our sponsors, Aaron Young (Operations Manager) and Kevin Mckay (Operations Supervisor) from Protection1 Security Solution and by Door Keepers. These requirements have been prioritized from 1 through 5, where 1 stands for high priority and 5 stands for low priority or future idea.

3.1 Door Status

3.1.1 Description: Depending upon the position of the door, the system shall update the status as open, close or in transition. When the door is opened completely then it should update its status to "open". When it is closed completely then it should update its status to "closed". When it is in the process of opening or closing, it should update its status to "in transition". If the door sensor does not trigger for some time during opening or closing, it should update its status to "jammed"

3.1.2 Source: Sponsor

3.1.3 Constraints: The time required to open varies on the opener model.

3.1.4 Standards: None

3.1.5 Priority: 1

3.2 Open Garage Door

3.2.1 Description: Garage door shall open when the homeowner sends the open signal over the Internet via the app or the web interface.

3.2.2 Source: Sponsor

3.2.3 Constraints: The home must be equipped with an Internet connection and a home router. The Wi-Fi signal to the garage must be strong or it should be hard wired.

3.2.4 Standards: None

3.2.5 Priority: 1

3.3 Close Garage Door

3.3.1 Description: Garage door shall close when the homeowner sends the close signal over the Internet via the app or the web interface.

3.3.2 Source: Sponsor

3.3.3 Constraints: The home must be equipped with an Internet connection and a home router. The Wi-Fi signal to the garage must be strong or it should be hard wired.

3.3.4 Standards: None

3.3.5 Priority: 1

3.4 Turn Lights On

3.4.1 Description: The light shall turn on when the homeowner sends light on signal over the Internet via the app or the web interface.

3.4.2 Source: Sponsor

3.4.3 Constraints: The home must be equipped with an Internet connection and a home router. The Wi-Fi signal to the garage must be strong or it should be hard wired.

3.4.4 Standards: None

3.4.5 Priority: 1

3.5 Turn Lights Off

3.5.1 Description: The light shall turn off when the homeowner sends light off signal over the Internet via the app or the web interface.

3.5.2 Source: Sponsor

3.5.3 Constraints: The home must be equipped with an Internet connection and a home router. The Wi-Fi signal to the garage must be strong or it should be hard wired.

3.5.4 Standards: None

3.5.5 Priority: 1

3.6 Security Notification

3.6.1 Description: The system shall notify the owner if an attempt is made to open the door without the signal sent from the unit. If the garage door is opened without being signaled from the system, the system should notify the homeowner through an email address that has been given by the homeowner.

3.6.2 Source: Sponsors

3.6.3 Constraints: Working email address must be entered.

3.6.4 Standards: All the email will be sent to only the address that has been set by the homeowner.

3.6.5 Priority: 2

3.7 System Log Record

3.7.1 Description: The system shall record all the activities occurred in the garage door. It should be stored in the database.

3.7.2 Source: Door Keepers

3.7.3 Constraints: None

3.7.4 Standards: Records will be kept according to time.

3.7.5 Priority: 2

3.8 Unit Operating Alert System

3.8.1 Description: The system shall buzzer a sound when the garage door will be operating from the unit.

3.8.2 Source: Door Keepers

3.8.3 Constraints: None

3.8.4 Standards: None.

3.8.5 Priority: 2

3.9 Multiple Garage Doors

3.9.1 Description: Multiple Smart Garage system should be handled from the same account. Just by adding new Smart Garage units, the homeowner should be able to control more than one garage door opener.

3.9.2 Source: Mr. O'dell

3.9.3 Constraints: None

3.9.4 Standards: Only one Smart Garage should be handled at one time. While one Smart Garage system is selected, it should not trigger the other one.

3.9.5 Priority: 5

3.10 Intercom Communication

3.10.1 Description: If someone arrives at the garage door and wants to enter, they could just press a button located in front of the garage door and talk with the homeowner through the intercom, which would be accessible by the homeowner through the smartphone app.

3.10.2 Source: Door Keepers

3.10.3 Constraints: Smartphone app should create a notification to inform the homeowner about the request from the intercom.

3.10.4 Standards: None

3.10.5 Priority: 5

3.11 Storing Video Surveillance

3.11.1 Description: When an event occurs in the garage, the Smart Garage should detect the event either by using motion sensor or by checking the status of the garage door, and store a short video from the camera on the system's SD card for future playback/review. Using motion sensor will allow the Smart Garage to check on any movements other than just relying on the status of garage door. After the SD card is full the system should override the old files with new ones.

3.11.2 Source: Door Keepers

3.11.3 Constraints: Requires a large amount of space to store video for long periods of time, but the system can only support up to 32GB SD card.

3.11.4 Standards: None

3.11.5 Priority: 5

3.12 Live Video Feed

3.12.1 Description: The system shall display the live video of the garage door via webpage and mobile app.

3.12.2 Source: Sponsors

3.12.3 Constraints: Requires good internet connection and may take all the data if running in mobile internet connection.

3.12.4 Standards: Should update the pictures of the garage door at least each second.

3.12.5 Priority: 1

3.13 Two-Step Verification

3.13.1 Description: The system shall ask for second verification code with the homeowner while opening or closing the garage door. The security code must match to perform the given functionality.

3.13.2 Source: Door Keepers

3.13.3 Constraints: None

3.13.4 Standards: The security code shall be 4 digit pin code.

3.13.5 Priority: 1

3.14 Notification Center

3.14.1 Description: The system shall allow the homeowner to access the system log records via webpage and mobile app.

3.14.2 Source: Door Keepers

3.14.3 Constraints: None

3.14.4 Standards: None

3.14.5 Priority: 2

3.15 Time Period Security System

3.15.1 Description: If the garage door is open for more than a certain period of time, then it should send an alert message to the homeowner. This is mainly to ensure the homeowner knows the door is still open. Homeowner should be able to change the time period according to their needs. Either to keep it for once, daily or weekly. Default time setup for the time period would be 15 minutes.

3.15.2 Source: Sponsor

3.15.3 Constraints: None

3.15.4 Standards: Owner should be able to change the time period as wished.

3.15.5 Priority: 3

3.16 Mobile Application Control

3.16.1 Description: The system shall be accessible via smartphone app. The smartphone app should have functions such as open, close, toggle light, and check status. Open should open the garage door. Close should close the garage door. Light should toggle the light. Check status should show the present status of the garage.

3.16.2 Source: Sponsor

3.16.3 Constraints: None

3.16.4 Standards: Each garage system should be operated from one user account only.

3.16.5 Priority: 1

3.17 Turn On/Off Notifications

3.17.1 Description: The system shall allow the homeowner to choose to receive email notifications. If notification is turned on, a working email address must be entered.

3.17.2 Source: Door Keepers

3.17.3 Constraints: None

3.17.4 Standards: To receive notifications, a working email address must be entered.

3.17.5 Priority: 2

3.18 Turn On/Off Buzzer

3.18.1 Description: The system shall allow the homeowner to choose to turn on or turn off the buzzer. If the homeowner will turn off the buzzer then the sound while operation through the unit should not sound.

3.18.2 Source: Door Keepers

3.18.3 Constraints: None

3.18.4 Standards: Buzzer should only make sound when it is turned on.

3.18.5 Priority: 2

3.19 Clear Log Records

3.19.1 Description: The system shall allow the homeowner to choose to clear the log records. All the records that have been recorded by the unit will be removed when the clear log record command is given.

3.19.2 Source: Door Keepers

3.19.3 Constraints: None

3.19.4 Standards: Once the data is deleted, it cannot be retrieved.

3.19.5 Priority: 2

4 Packaging Requirements

The complete package from the Door Keepers includes four main components: hardware, software, quick setup instructions, and a full user manual. Our hardware will be the microcontroller, door sensors, camera, and other internal components. The software will contain the pre-installed operating system and PDF versions of the quick setup guide and manual. From a homeowner's perspective, they will only see a box with a bottom mounted camera, but all of our components will be inside. Our Smart Garage unit will be mounted above the garage door opener and will have connections from the unit directly the garage door opener. The unit will also have the option to be connected via an Ethernet cord to the home network, but Wi-Fi will be the primary link of communication.

4.1 The Recovery Disc

4.1.1 Description: The system's software shall be delivered on a CD. This serves as a backup for the provided system software, quick setup guide, and manual.

4.1.2 Source: Door Keepers

4.1.3 Constraints: The homeowner must use a PC, with a PDF viewer program installed onto it, to view the CD and make use of its contents.

4.1.4 Standards: None

4.1.5 Priority: 1

4.2 Parts Listing

4.2.1 Description: The product shall contain a webcam, magnetic door sensors, 100 feet of wire for the sensors, mounting bracket with screws, a power adaptor, and a microcontroller. The microcontroller will come fully assembled and pre-loaded with software. All of the components will be packaged in a hard enclosure which will prevent dust from entering. The unit will also be able to withstand high and low temperatures from weather.

4.2.2 Source: Door Keepers

4.2.3 Constraints: This requirement may be subject to additions due to what the microcontroller requires. The pre-assembled controller will include an SD Card with software, a speaker, and a Wi-Fi adapter at the very least.

4.2.4 Standards: All associated hardware required for the system to operate will be included.

4.2.5 Priority: 1

4.3 Full User Manual

4.3.1 Description: The final product will include a printed user manual in English. The User Manual will mention all full details of everything our product can do and how to install it to the garage system.

4.3.2 Source: Door Keepers

4.3.3 Constraints: The language used to describe how to use the product should be understandable by a novice for our hardware.

4.3.4 Standards: The user manual will contain instructions over all aspects of the user software and app. Any major updates to the software or app later on will necessitate a new version of this manual, to be provided online as a PDF document.

4.3.5 Priority: 1

4.4 Quick User Guide

4.4.1 Description: The final product will include a printed quick user guide prepared in English.

4.4.2 Source: Door Keepers

4.4.3 Constraints: This quick user guide will be kept to a single 8.5” x 11” page to briefly explain the setup processes to the customers.

4.4.4 Standards: None

4.4.5 Priority: 1

4.5 Smartphone Application

4.5.1 Description: The Smartphone application will be able to be downloaded via a web URL from the Smart Garage website or the homeowner will be able to find it on the Google Play Store.

4.5.2 Source: Door Keepers

4.5.3 Constraints: The user must have an Android phone with Google Play Store installed.

4.5.4 Standards: None

4.5.5 Priority: 1

5 Performance Requirements

This section covers the time response, performance quality and installation time for the Smart Garage.

5.1 Garage Response Time

5.1.1 Description: The system shall be responsive to commands given by the homeowner within 10 seconds.

5.1.2 Source: Door Keepers

5.1.3 Constraints: The system and smartphone must have internet access during this period.

5.1.4 Standards: None

5.1.5 Priority: 1

5.2 Status Update Time

5.2.1 Description: The system shall update the status of the garage door as changes occur. These updates should propagate to the website and app in less than 15 seconds. Notify if time-outs occur.

5.2.2 Source: Door Keepers

5.2.3 Constraints: Both the product and the app need to be connected to the internet.

5.2.4 Standards: None

5.2.5 Priority: 1

5.3 Camera

5.3.1 Description: The system shall provide a live video stream of the garage door whenever the owner desires. It should take less than 5 seconds to initiate and send the stream to the app or system web page. Live video streaming will not record the video

5.3.2 Source: Door Keepers

5.3.3 Constraints: Both the system and app need an internet connection. The camera must face towards the garage door in order to provide the homeowner with a clear view of the door. Data transmission speeds will vary, so the stream will need to be compressed for faster streaming.

5.3.4 Standards: None

5.3.5 Priority: 1

6 Safety Requirements

The Smart Garage product will be developed for a plug and play style installation. Therefore, many security flaws will be eliminated, but there will still be a few safety restrictions that we must account for.

6.1 Unit Installation

6.1.1 Description: The Smart Garage system shall connect to existing in-home garage door opener systems. During this process, the homeowner may make mistakes which may result as electrical failures or shock. To prevent this from happening, descriptions with pictures will be provided in the Quick Installation Guide and User Manual.

6.1.2 Source: Door Keepers

6.1.3 Constraints: The existing in-home garage door opener may not be compatible with our smart garage system for terminal connection.

6.1.4 Standards: None

6.1.5 Priority: 1

6.2 Electrical Safety

6.2.1 Description: The Smart Garage system will have warnings in the user manual and labels on the unity about possible shock risks.

6.2.2 Source: Door Keepers

6.2.3 Constraints: None

6.2.4 Standards: None

6.2.5 Priority: 1

7 Maintenance and Support Requirements

The maintenance and support requirements for the Smart Garage System are outlined in this section. It explains how to maintain a fully functioning system to the homeowner. The following requirements shall be met to make the system maintenance and upgrade easy and effective.

7.1 User Manual

7.1.1 Description: The User Manual shall include sections with step by step procedures on how to troubleshoot, maintain, and upgrade both hardware and software systems.

7.1.2 Source: Door Keepers

7.1.3 Constraints: There may be specific knowledge required for hardware/software upgrading. This will be detailed in the manual

7.1.4 Standards: None

7.1.5 Priority: 1

7.2 Installation/System Setup

7.2.1 Description: The system shall not require major modification of existing garage door openers or home networks. Being an add-on type device, it should not necessitate an overhaul of existing hardware.

7.2.2 Source: Door Keepers

7.2.3 Constraints: None

7.2.4 Standards: None

7.2.5 Priority: 1

7.3 Hardware Support/Upgrades

7.3.1 Description: Homeowners shall be able to make changes to the hardware. The microcontroller chosen will be an open platform for them to make modifications, upgrade, or fix parts.

7.3.2 Source: Door Keepers

7.3.3 Constraints: Specific knowledge of the hardware will be required of the homeowner to accomplish this, namely in the kind of microcontroller we will employ, the software behind it and its parts, and parts that are compatible with the platform.

7.3.4 Standards: Troubleshooting information on certain hardware parts should be included in the manual.

7.3.5 Priority: 2

7.4 Software Support/Upgrades

7.4.1 Description: Software updates shall be usable and deployable by the homeowners.

7.4.2 Source: Door Keepers

7.4.3 Constraints: Software upgrades will be initiated from the internal website only.

7.4.4 Standards: The configuration page should have a “check update” button. If there is a new version, it will show the version and ask to update. Upon accepting the update, it will download the package, update files, and reboot the unit.

7.4.5 Priority: 1

7.5 Source Code

7.5.1 Description: The operating system used and software developed shall be open sourced for inspection and modification by anyone. The source will be documented via comments for easy readability.

7.5.2 Source: Door Keepers

7.5.3 Constraints: Knowledge of the programming languages used in construction will be required to understand and modify the source.

7.5.4 Standards: GPL or other free-software licensing will be utilized. Comments will include describing functions with their inputs and outputs, external source URL, links, and citations on code not produced by the team, and any other helpful information inside the functions.

7.5.5 Priority: 2

8 Other Requirements

These requirements do not fit under any of the previous sections, but detail specific requirements about miscellaneous aspects of the product.

8.1 Customer Facing Interface

8.1.1 Description: The entire system shall be usable without needing to use the terminal or command prompt for any setup or feature. All features and operations shall be functional via the web interface. Most operations should be usable via the smartphone app.

8.1.2 Source: Door Keepers

8.1.3 Constraints: Must be entirely usable from a web interface and the app.

8.1.4 Standards: The app should have basic functionality, such as adding the home unit, open and close door, turn on light, check status, and view camera. The web interface should have all those features along with extra setup options, such as inputting a noip.com login, setting up user account, and checking all components to verify they all work properly

8.1.5 Priority: 1

8.2 Temperature Tolerance

8.2.1 Description: The system shall be able to withstand temperatures common to unheated or uncooled garages.

8.2.2 Source: Door Keepers

8.2.3 Constraints: We need to find the temperature breaking points of our hardware. A range of -20F to 140F would be ideal.

8.2.4 Standards: None

8.2.5 Priority: 5

8.3 Customer Device Account

8.3.1 Description: The system shall enable the creation of a user account for the homeowner to secure access to the Smart Garage.

8.3.2 Source: Door Keepers

8.3.3 Constraints: Only one username and password per device. Multiple users of one Smart Garage with different access levels may be troublesome.

8.3.4 Standards: None

8.3.5 Priority: 1

8.4 Hardware System Setup

8.4.1 Description: To setup hardware, homeowner may have to spend couple of hours. To function perfectly, the product have to be mounted near the homeowner's garage door opener and connected to garage door opener. It also needs to be connected to the home's router either hardwired or wireless. Sensors needs to be positioned perfectly at top and bottom of the garage frame and one at the garage door. This should take up to 1 and a half hours to 2 hours to setup the hardware components.

8.4.2 Source: Door Keepers

8.4.3 Constraints: The homeowner must have internet service at home. The user will also need good wireless reception in the garage or have to run a long cable from their router to the product. Additional time may be required due to opener's positioning, troubleshooting the switches, or hard-wired cabling.

8.4.4 Standards: None

8.4.5 Priority: 1

8.5 Software System Setup

8.5.1 Description: To access the system through the internet, the homeowner will have to setup an account, which may take up to 20 minutes. To get access to the system, the homeowner will have to download the Smart Garage app, which would be available in

Play Store for Android Devices. For the Smart Garage app, the homeowner will first need to setup the Smart Garage via the web interface. To download, install and setup account for the Smart Garage app it may take up to 30 minutes. So, to setup the software it may take 50 minutes.

8.5.2 Source: Door Keepers

8.5.3 Constraints: The homeowner must have internet service at home. The user will also need good wireless reception in the garage or have the system hardwired.

8.5.4 Standards: None

8.5.5 Priority: 1

8.6 Garage Door Open and Close Time

8.6.1 Description: This would depend on the type of garage door, but it should not take more than 45 seconds for a normal home-car garage. If it takes more than 45 seconds to open the door, then update the status to jammed and provide a new photo from the embedded camera.

8.6.2 Source: Door Keepers

8.6.3 Constraints: The openers themselves have their own open/close times that we must take into account.

8.6.4 Standards: None

8.6.5 Priority: 1

8.7 Browser Support

8.7.1 Description: The user interface should be accessible from all major browsers including Internet Explorer 9.0+, Mozilla Firefox, and Google Chrome.

8.7.2 Source: Door Keepers

8.7.3 Constraints: The browser must support Motion JPEG, which all current versions of Firefox, Chrome, and IE 9.0+ support as of 4/4/2014.

8.7.4 Standards: None

8.7.5 Priority: 1

9 Acceptance Criteria

This section lists the features and functions of Smart Garage System that the Door Keepers should deliver to make sure the sponsor and users be satisfied and accept our product. It contains all deliverables or requirements that are expected to be delivered by the Door Keepers.

9.1 The system must be able to update and display the status of the garage door.

9.1.1 Requirement(s) addressed: Requirement 3.1 Door Status: Depending upon the position of the door, the system must be able to update the status as “Open,” “Close,” “In Transition” or “Jammed”.

9.1.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by using the Smart Garage app or the system’s web interface. For example, the sponsor might want to know the current status of the garage door, so they push the button “Status” on the app and visually ensure the response is correct.

9.2 The system must have control operations for the garage door over the Internet.

9.2.1 Requirement(s) addressed: Requirement 3.2 Door Control: Garage door must be operational over the Internet. The owner must be able to operate the system over the Internet.

9.2.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by testing over the Internet. For instance, the sponsor can click a menu button “Close” on the app or system’s web interface. After receiving the command from the sponsor, the door will close and be visually verifiable.

9.3 The system must have control operations for the Smart Garage's light over the Internet.

9.3.1 Requirement(s) addressed: Requirement 3.3 Turn Lights On and Requirement 3.5 Turn Lights Off: Lights must be operational over the Internet. The user must be able to turn lights on or off over the Internet.

9.3.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by testing over the Internet. For instance, the sponsor can click a menu button “Turn on” or “Turn off” on the app or system web interface. After receiving the command from the sponsor, the light should turn on or off and be visually verifiable.

9.4 The team must deliver a mobile application to control the garage door.

9.4.1 Requirement(s) addressed: Requirement 3.16 Mobile Application Control: The system must operate the garage door using a smartphone app.

9.4.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by using the app. The smartphone app must include features such as open, close, toggle light, and check status. For example, the customer can press a menu button “close” on the app. After the customer pressed the button the system recognizes and executes the command and the garage door will be closed, and can be visually verified.

9.5 The system must produce a notification when the garage door stays open for more than a fixed period of time.

9.5.1 Requirement(s) addressed: Requirement 3.15 Time Period Security System: The system must notify the sponsor whenever the garage door has stayed open for more than a certain period of time. The sponsor can manually set the amount of time period when the system will produce a notification to the sponsor. The system has a default time of 15 minutes.

9.5.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by testing the operation on the garage door. For instance, the sponsor can set a certain time period beyond which the system will produce a notification if the garage door stayed open. The sponsor can then open the garage door using the app and wait till the time is up. When the set time is reached, the system recognizes the door is still open and produces a notification to the app.

9.6 The system must produce a notification when an attempt is made to open the garage door without the signal sent from the unit.

9.6.1 Requirement(s) addressed: Requirement 3.6 Security Notification: The system must notify the sponsor whenever an attempt is made to open the garage door without the system command.

9.6.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by testing the operation on the garage door. For instance, the sponsor may try to open the

door without using the system. Then the system should recognize what is happening on the garage door and notify the sponsor about it.

9.7 The system must provide live video of the garage door.

9.7.1 Requirement(s) addressed: Requirement 3.12 Live Video Feed: The system must display the live video of the garage door via webpage and mobile app.

9.7.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by visually verifying that the picture being shown in the web and app are same from the garage door.

9.8 The system must store the events occurring in the Smart Garage system and display them through the website or mobile app.

9.8.1 Requirement(s) addressed: Requirement 3.7 System Log Record and Requirement 3.14 Notification Center: The system must store all the events occurring in the Smart Garage system and display them through the website or mobile app.

9.8.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by performing different events in the Smart Garage system and checking the Notification Center for these records.

9.9 The system must ask for security code for performing certain activities.

9.9.1 Requirement(s) addressed: Requirement 3.13 Two Step Verification: The system must ask for security code when the open or close door functions are performed.

9.9.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by setting up the pin number and trying to open and close the garage door by entering the same pin number.

9.10 The system must sound a buzzer when the door is triggered from the unit.

9.10.1 Requirement(s) addressed: Requirement 3.8 Unit Operating Alert System: The system must sound a buzzer when the garage door is activated by the unit.

9.10.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by opening or closing the garage door and listening for the buzzer.

9.11 The system must provide the option to turn on or off notifications.

9.11.1 Requirement(s) addressed: Requirement 3.17 Turn On/Off Notification: The system must allow the homeowner to choose to receive or not to receive email notifications.

9.11.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by turning notification on or off and checking if emails were sent or not sent during notification events.

9.12 The system must provide the option to turn on or off the buzzer.

9.12.1 Requirement(s) addressed: Requirement 3.18 Turn On/Off Buzzer: The system must allow the homeowner to choose to turn on or off the buzzer.

9.12.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by opening or closing garage door and listening for the buzzer sound. If buzzer is turned on then it should be triggered.

9.13 The system must provide an option to clear the log record.

9.13.1 Requirement(s) addressed: Requirement 3.19 Clear Log Record: The system must allow the homeowner to clear the log record.

9.13.2 Verification Procedure: The sponsor will be able to verify that the requirement is met by clicking the clear log record button and opening the notification center.

10 Use Cases

This section shows possible uses of the Smart Garage smartphone app and system. "This Use Case Begins With" has been abbreviated to "TUCBW" and "This Use Case Ends With" with "TUCEW" for this section.

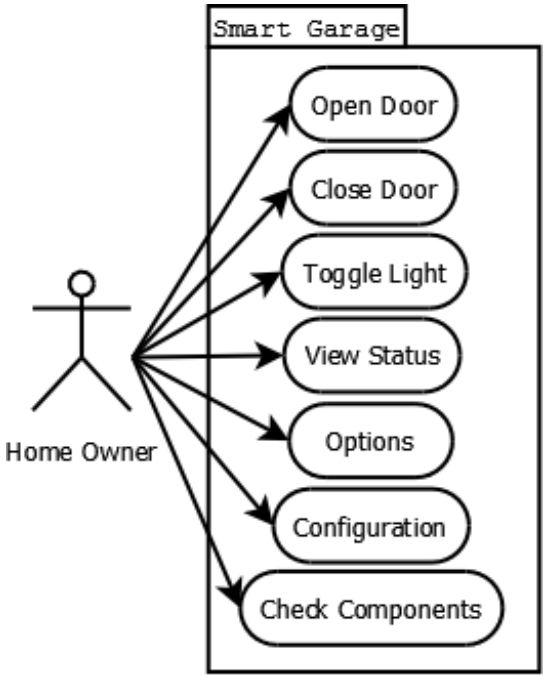

Figure 3: Smart Garage Use Case Diagram

10.1 Smart Garage App Use Cases

The homeowner uses the smartphone app to control the Smart Garage unit.

10.1.1 Open Door

10.1.1.1 **Scenario:** The home owner taps the "Open Door" button. The garage door opener tied to the app will now open its door.

10.1.1.2 **Actor:** Home Owner

10.1.1.3 **TUCBW:** The home owner taps the "Open Door" button.

10.1.1.4 **TUCEW:** The home owner opens his/her garage door.

10.1.2 Close Door

10.1.2.1 **Scenario:** The home owner taps the "Close Door" button. The garage door opener tied to the app will now close its door.

10.1.2.2 **Actor:** Home Owner

10.1.2.3 **TUCBW:** The home owner taps the "Close Door" button.

10.1.2.4 **TUCEW:** The home owner closes his/her garage door.

10.1.3 Toggle Light

10.1.3.1 **Scenario:** The home owner taps the "Toggle Light" button. The garage door opener tied to the app will now toggle its light on or off.

10.1.3.2 **Actor:** Home Owner

10.1.3.3 **TUCBW:** The home owner taps the "Toggle Light" button.

10.1.3.4 **TUCEW:** The home owner turns on or off his/her garage door opener's light.

10.1.4 View Status

10.1.4.1 **Scenario:** The home owner taps the "View Status" button. The app will display a stream from the unit's camera on screen and get current status from the unit's sensors.

10.1.4.2 **Actor:** Home Owner

10.1.4.3 **TUCBW:** The home owner taps the "View Status" button.

10.1.4.4 **TUCEW:** The home owner is shown a stream from the garage cam and updated current status.

10.1.5 Options

10.1.5.1 **Scenario:** The home owner taps the "Options" button. The app will display a screen to alter configuration settings for the unit. These settings include web or IP address of the unit, port, time period for alert, and username and password of the unit.

10.1.5.2 **Actor:** Home Owner

10.1.5.3 **TUCBW:** The home owner taps the "Options" button.

10.1.5.4 **TUCEW:** The home owner is given a screen of alterable configuration settings.

10.1.6 Configuration

10.1.6.1 **Scenario:** The home owner clicks on the "Configuration" button. A page will display with editable entries to customize aspects of the unit. These aspects include: username and password for the unit, username and password of a dynamic DNS forwarding service (ex. no-ip.com), home wireless settings, time period for alert, and future configuration settings.

10.1.6.2 **Actor:** Home Owner

10.1.6.3 **TUCBW:** The home owner clicks on the "Configuration" button.

10.1.6.4 **TUCEW:** The home owner can configure their unit.

10.1.7 Check Components

10.1.7.1 Scenario: The home owner clicks on the "Check Components" button. A page containing current component information is shown. Information includes: wireless activity, sensor status and data, camera status and stream view, and future component statuses.

10.1.7.2 Actor: Home Owner

10.1.7.3 TUCBW: The home owner clicks on the "Check Components" button.

10.1.7.4 TUCEW: The home owner gains access to statuses and data of all the connected components.

11 Feasibility Assessment

Door Keepers have conducted some analysis on our overall strengths and weaknesses as a group against our project idea. In this section, we will use the following six analysis sections to determine how well we will complete this project: scope analysis, research analysis, technical analysis, cost analysis, resource analysis, and schedule analysis. A final section, conclusion, will interpret our projected results and provide a summary.

11.1 Scope Analysis

We have seen many other individuals accomplish this sort of project in weeks as there are many open sourced example projects online. Nearly all however, are made with just one user in mind: the tinkerer. Our aim is to try and make this more accessible by providing a self-contained kit that can be installed relatively quickly by someone who is at least comfortable with electronics. With a small feature set, this serves to provide a quick installation with minimal possible problem points.

Our sponsor, Protection1, also was in agreement that this device should not be very complicated and have a small feature set to start. They had several commercial products with similar sorts of monitoring, but were completely tied with specific big home security systems.

We, however, have not dealt with any microcontrollers or their associated hardware and software. We also had other features planned at the beginning, such as an intercom outside the door with a camera, use of night-vision/IR cameras, recording/streaming video/audio, temperature gauge, small LCD display for on-unit statuses, and even some image processing to determine the door status without needing the door sensors. Some of these ideas are outside our capability for now and would complicate the product further while others are saved for future implementation.

11.2 Research

Our most basic functions are in setting up the microcontroller as a web server complete with an internal user interface, getting data out of the sensors, using the hardware (camera, opener hardware, USB wireless stick, etc), and app development.

11.2.1 Web Server: A couple of us are familiar with web development, though we will need to create interfaces to help ease and automate setup. The homeowner should never see the underlying OS or need to use a terminal/command prompt, and being able to achieve this should be within our grasp. We will also need to handle storing captured pictures, serving them, and possibly keeping track of events with a small database (times when system was used, actions taken, etc).

11.2.2 Hardware: None of us are familiar with microcontroller hardware, but we are aware of their capabilities and the online communities around such devices. We chose this due to

how many great projects there are for various microcontrollers and how available code is for specific little parts. The cost was also a factor; most are very inexpensive to replace should we break something. With the shear amount of code and sites dedicated to microcontrollers, our development and setup is significantly reduced, but we still require research in the various I/O ports available for use and the kinds of software we need to make for it.

11.2.3 App Development: We have little experience developing apps, but the nature of the current requirements would enable a faster creation time. With few screens and interactions, we can concentrate on the more complicated tasks that take place on the unit itself. We will first design and build the web interface and then adapt it to an app. Android is our target for right now, but the website will be usable on all platforms. This will likely entail the use of Java and certain web technologies to communicate with the Smart Garage system. We all have experience in Java, so we are mostly missing, and must research, the web and app tech.

11.3 Technical Analysis

The idea of the Smart Garage system is not something new. There are dozens of small projects based around this idea with varying amounts of interaction, accessibility, and usability, along with a small number of commercial products that tie in with larger home systems. Our aim is to make it more accessible, customer owned (not on a central server), and very simple to use.

The main challenges are in getting the hardware and software to all communicate. The microcontroller will need to interface with an existing garage door opener, controlling its opening, closing, and lights, and have sensors detecting door-action. This is the main part of this project: getting control over the opener. This will involve a lot of research in the I/O ports of our chosen micro-controller, as well as the available ports on the opener hardware. Web server development for the microcontroller is the second big part, as it must report statuses, serve pictures or video, and provide configuration support in an easy to use interface. There are many sample projects for web servers and interfaces for these devices, which can speed up development a bit and then create the pages and database how we think we need it. The third part is the app, which is more an extension of the web server's user page, and the third big part of the package. We will need to run through some sample app developments in order to get our bearings in the new environment.

11.4 Cost Analysis

Door Keepers have been allotted $800 for expenses by UTA for the project. By using a small, programmable microcontroller, we save on cost since most are very open, documented, and have cheap accessories. The main cost will be in the garage door opener hardware to test with, and finding one that is easy to interface with. All software will either be free (gratis, GPL, freedom) open sourced distributions or written by our team.

Item	Quantity	Price	Total
Microcontroller Kit Kit Includes: Buzzer, LCD Screen, Enclosure, Breadboard, Wires, Unit, Power, Relay	1	varies: $60-$80	$60-$80
Web Camera	1	$30	$30
Garage Door Opener	1	varies: $150-$300	$150-$300
Door Sensor (reed switch type)	2	$15	$45
Wires (for sensors/hook ups)	100ft	$0.10/ft	$10
Publishing on Google Play	1	$25	$25
		Total:	**$320-$490**

Table 2: Cost Estimation

We are well within budget for this project, though this is only an opening estimate based on current prices on Amazon. There are various Microcontroller Kits that include a case, power adaptor, USB wireless adapter, SD card, and various other components. We also need to settle on one garage door opener to have a specific dollar amount for the budget. Publishing the app to the Google Play Store is also factored in as it is an easy platform to get the app out and installed quickly while ensuring users have the newest version of the app.

11.5 Resource Analysis

The Door Keepers is comprised of a diverse group of students in the senior design class at The University of Texas at Arlington. On our team we have 2 computer science members (Anup, Adrian), one computer engineer (Santosh), and one software engineer (Wasyhun). Santosh will be primarily in charge of the hardware components, Anup will be in charge of web development, Adrian will lead Android development, and lastly Wasyhun will assist in linking the hardware and software components together. Even though each member has a primary role in each field, they will all be assisting each other to get this project completed.

11.6 Schedule Analysis

To give us an estimate on how long it will take to complete the Smart Garage, our team has used multiple estimation techniques including function points, Jones First Order Estimation, and COCOMO II to give us a brief estimate on completion time.

Low, Medium, and High reflect how hard we think it will be to access these components.

External Inputs

- Garage Door Opener - Unit (Low)
- Smart Garage - Sensors (Low)
- Database - Username/Password Storage (Low)
- Phone - Touch and Login Interface (Low)
- Phone - Open/Close Door Control (Low)
- Phone - Turn On/Off Light (Medium)
- Phone - Options Page (Low)
- Web - Maintenance and Setup for Smart Garage (Medium)

External Outputs

- Garage Door Opener - Door (Low)
- Smart Garage - Alarm Speaker (Medium)
- Garage Door Opener - Light (Low)
- Phone - Status Of Garage From Smart Garage (Low)

External Inquiry

- Phone - Login (Low)
- Phone - Open/Close Garage (Low)
- Phone - Turn Light On/Off (Low)
- Phone - View Snapshot/Video (Low)
- Web - Login (Low)
- Web - Options Page (Low)

Internal Logical Files

- Phone - Saved Snapshots/Videos (Low)
- Phone - User Account Auto Login (Low)
- Phone - GUI (Low)

External Interface Files

- Smart Garage - Network IP (Medium)

Function Type	Low Complexity	Medium Complexity	High Complexity	Functional Point Total
Inputs	6*3	2*4	0*6	26
Outputs	3*4	1*5	0*7	17
Inquiries	6*3	0*4	0*6	18
Logical Internal Files	3*7	0*10	0*15	21
External Interface Files	0*5	1*7	0*10	7
Unadjusted Total				**89**
Adjustment Factor				**0.92**
Adjusted Total				**81.88**

Table 3: Function Points

Characteristics	Effort (0-5)
Data Communications	1
Distributed Data Processing	2
Performance	2
Heavily Used Configuration	2
Transaction Rate	4
Online Data Entry	1
End User Efficiency	4
Online Update	2
Complex Processing	1
Reusability	2
Installation Ease	2
Operation Ease	1
Multiple Sites	1
Facilitate Change	2
Total	**27**
Value Adjustment Factor	**0.92**

Table 4: Influence Multipliers

To achieve the Value Adjustment Factor, we have used the following formula:

VAF = (Total*.01)+.65

This will help us estimate the Jones First Order to the best of our ability.

Best Case	Average Case	Worst Case
81.88^.43	81.88^.45	81.88^.48
6.65 Calendar Months	7.26 Calendar Months	8.29 Calendar Months

Based on the results from Jones' First Order Estimation, we can see that this will be a lengthy project that will involve a lot of smaller functions. Based on our estimation points we believe that this project is doable in the maximum of 8.29 months, but will require full dedication from our team. Our team is primarily busy during the school semester, but in the summer we will be able to put more dedication into our project to help complete it on time. We have based our estimation exponents on a systems estimation and plan on accounting for worst-case.

We choose the systems estimation because the Smart Garage is both software and hardware, and we account for worst-case because The Door Keepers are students who have no industry experience and little hardware experience

COCOMO II Estimation

SCALE FACTORS:
PREC - Precedentedness : Nominal - 3.72
FLEX - Development Flexibility : High - 2.03
RESL - Architecture/Risk Resolution : Nominal - 4.24
TEAM - Team Cohesion : High - 2.19
PMAT - Process Maturity : High - 3.12

Exponent derived from the five Scale Drivers:
E = B + 0.01 X (all scale factors added up)
B = 0.91 (constant)
E = 0.91 + 0.01(15.3)
E = 1.063

Schedule equation exponent derived from the five Scale Drivers:
SE=0.3106

COST FACTORS:
-ACAP – Analyst Capability Cost Driver - HIGH
-APEX – Application Experience Cost Driver - NOMINAL
-PCAP – Programmer Capability Cost Driver - HIGH
-PLEX – Platform Experience Cost Driver - LOW
-LTEX – Language and Tool Experience Cost Driver - HIGH
-PCON – Personnel Continuity Cost Driver - NOMINAL
-TOOL – Use of Software Tools cost driver - HIGH

-SITE – Multisite Development Cost Driver - VERY HIGH
-SCED – Development Schedule Cost Driver - NOMINAL
-TIME – Execution Time Constraint Cost Driver - NOMINAL
-STOR – Main Storage Constraint Cost Driver - NOMINAL
-PVOL – Platform Volatility Cost Driver - LOW
-RELY – Required Reliability Cost Driver - VERY LOW
-DATA – Database Size Cost Driver - NOMINAL
-CPLX – Product Complexity Cost Driver - LOW
-RUSE – Require Reusability Cost Driver - LOW
-DOCU – Documentation Match to Life-Cycle Needs - NOMINAL

Effort Adjustment Factor derived from the Cost Drivers:
EAF = 0.3386

COCOMO II Equations:
Effort= 2.94*EAF*(KSLOC)^E
Effort= 2.94*0.3386*(1.5)^1.063= 1.53 person-months

Duration= 3.67*(Effort)^SE
Duration= 3.67*(1.53 person-months)^0.3106= 4.18 months

11.7 Conclusion

According to COCOMO II, we can see that the estimated duration for this project is only 4.18 months. This is very capable of completing with our group of four part time developers. If we divide this number by 4, it would come out to 1.045 months per person which would be a full-time month. As we are students we cannot dedicate full time to this project, but our duration to work on the project is just under 8 months per person to complete the task.

With the estimates from both Jones' First Order and COCOMO II, we can confirm that this project is feasible to complete by our team within the required deadlines.

12 Future Ideas

These are the requirements that have been prioritized for future consideration.

12.1 Customer Requirements 3.10: Intercom Communication

12.1.1 Requirement Description: The Intercom Communication specified in Customer Requirements 3.10 must be able to connect an intercom outside the garage door with the app.

12.1.2 Constraint: Door Keepers does not have sufficient skills and knowledge to implement intercom communications and we may not have enough time to complete this task.

12.2 Other Requirements 8.2: Temperature Tolerance

12.2.1 Requirement Description: The Temperature Tolerance specified in Other Requirements 8.2 must be able to withstand temperatures common to unheated/uncooled garages.

12.2.2 Constraint: The temperature breaking points of the hardware is unknown. The team will need more research and knowledge to further weather-proof the product.

12.3 Customer Requirements 3.11: Storing Video Surveillance

12.3.1 Requirement Description: The Storing Video Surveillance specified in Customer Requirements 3.11 must be able to store the video when an event occurs in the garage.

12.3.2 Constraint: It may require a lot of storage space. Most microcontrollers only support 32GB SD Cards at maximum, which can be a limiting factor. Depends entirely on how small the bytes used per second of video can be.

12.4 Customer Requirements 3.9: Multiple Garage Doors

12.4.1 Requirement Description: The Multiple Garage Doors specified in Customer Requirements 3.9 must be able to operate more than one Smart Garage System.

12.4.2 Constraint: This project has only budget of $800. Expenses expected for operating only one Smart Garage is about $475.00. Economically, this requirement may not be feasible.